PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of

Chang et al. (SANDP040) Conf. No. 2329

Serial No. 10/678,893 Group Art Unit: 2186

Filed: October 2, 2003 Examiner: Tsai

For: Hybrid Implementation for Error Correction Codes Within a Non-Volatile Memory System

APPELLANTS' BRIEF

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Appellants respectfully present their brief in support of their appeal of the final rejection of claims in this case. The Notice of Appeal was filed on January 19, 2007, as indicated by the automated electronic receipt from the Patent and Trademark Office.

Real Party in Interest

The real party in interest in this application is SanDisk Corporation.

Related Appeals and Interferences

None.

Status of the Claims

Claims 1, 3 through 11, 13 through 21, 23 through 28, 30 through 34, 38, 40 through 42, 44, and 45 were finally rejected in the Office Action of September 19, 2006. Claims 3, 4, 6

through 11, 13, 16 through 21, 23, 26 through 28, and 30 through 34 are the subject of the present appeal.

Status of Amendments

An amendment was presented after the final rejection on December 19, 2006. In that amendment, claims 3, 4, 7 through 9, 11, and 21, were proposed to be amended, and claims 1, 5, 14, 15, 25, 38, 40 through 42, 44, and 45 were proposed to be canceled. As indicated in the Advisory Action of January 12, 2007, this amendment was entered, but the accompanying arguments with that amendment were not found to be persuasive.

Summary of the Claimed Subject Matter

Independent method claim 4 is directed to a method of storing data within a plurality of blocks in a non-volatile memory, such as a flash memory.[1] An indicator of whether the block is a reclaimed block[2] is associated with each block of the non-volatile memory, and this indicator is interrogated (512), prior to the storing of data in a given block.[3] By way of example, a "reclaimed" block is a block that was once determined to be unusable, but which has been reclaimed, for example through the use of a rigorous testing process.[4] Upon the value of the indicator meeting a criterion, for example indicating that the block is not a reclaimed block, data is encoded (528) according to a first error detection algorithm, such as a 1-bit ECC algorithm, and the encoded data is written (532) into the block.[5] On the other hand, if the value of the indicator does not meet that criterion, data is encoded (516) according to a second error detection algorithm (*e.g.* a 2-bit ECC algorithm) that has a higher error detection capability than the first algorithm (*e.g.,* capable of detecting two failed bits rather than one failed bit), and this encoded data is written (520) into the block.[6] Claim 4 expresses this method in terms of a first and a second block, where the first and second blocks are written with data encoded according to the

[1] Specification of Appl. S.N. 10/678,893, page 9, lines 10 through 23; Figures 1a through 1c, elements 120, 124, 174.
[2] Specification, *supra,* page 20, lines 10 through 21; Figure 5a.
[3] Specification, *supra,* page 20, lines 18 through 28; Figure 5a.
[4] Specification, *supra,* page 18, line 10 through page 19, line 9.
[5] Specification, *supra,* page 20, line 25 through page 21, line 8; Figure 5a.
[6] Specification, *supra,* page 20, lines 19 through 23; Figure 5a.

first and second error detection algorithms, respectively, to convey the limitation that different blocks within the same array can be written with different error detection algorithms, depending on the value of their respective indicators.

Independent method claim 6 is also directed to a method of storing data within a plurality of blocks in a non-volatile memory, such as a flash memory.[7] According to claim 6, an erase count value that is associated with each block of the non-volatile memory, is compared (212) against a threshold error count value.[8] Upon the indicated erase count for the block being less than the threshold value, data is encoded (224) according to a first error detection algorithm, and the encoded data is written (228) into the block.[9] If the erase count indicator is not less than the threshold value, data is encoded (216) according to a second error detection algorithm that has a higher error detection capability than the first algorithm (*e.g.,* capable of detecting two failed bits rather than one failed bit)[10], and this encoded data is written (220) into the block.[11] Claim 6 expresses this method by reciting its method steps for a first block, and repeating those steps for another block, with the result being that the two blocks are written with data encoded according to the first and second error detection algorithms, establishing that different blocks within the same memory array can be written with different error detection algorithms according to the claimed method.

Independent method claim 16 is directed to a method of reading data that has been stored in a block within a non-volatile memory, such as a flash memory.[12] According to this method, upon identifying (244) a particular block to be read,[13] an indicator such as an erase count[14] associated with that block is compared (252) with a criterion.[15] Upon the value of the indicator meeting a criterion, for example being below a threshold erase count, data is decoded (256)

---

[7] Specification, *supra,* page 9, lines 10 through 23; Figures 1a through 1c, elements 120, 124, 174.
[8] Specification, *supra,* page 14, lines 9 through 26; Figure 2a.
[9] Specification, *supra,* page 15, lines 6 through 11; Figure 2a.
[10] *See* claim 13.
[11] Specification, *supra,* page 15, lines 13 through 18; Figure 2a.
[12] Specification, *supra,* page 9, lines 10 through 23; Figures 1a through 1c, elements 120, 124, 174.
[13] Specification, *supra,* page 15, lines 20 through 29; Figure 2b.
[14] Specification, *supra,* page 16, lines 1 through 3; Figure 2b.
[15] Specification, *supra,* page 16, lines 5 through 7. .

according to a first error detection algorithm, such as a 1-bit ECC algorithm.[16] On the other hand, if the value of the indicator does not meet that criterion, data is decoded (260) according to a second error detection algorithm (*e.g.* a 2-bit ECC algorithm) that has a higher error detection capability than the first algorithm (*e.g.,* capable of detecting two failed bits rather than one failed bit).[17]

Independent claim 21 is directed to a memory system that includes a non-volatile memory with a plurality of blocks in an array (124; 174).[18] This memory system includes a memory area that stores various program code devices.[19] As recited in claim 21, these code devices include code devices that obtain an indicator of whether that block is a reclaimed block[20]. Code devices are provided to encode the data according to a first error detection algorithm, such as a 1-bit ECC algorithm, or a second error detection algorithm (*e.g.* a 2-bit ECC algorithm) that has a higher error detection capability than the first algorithm (*e.g.,* capable of detecting two failed bits rather than one failed bit), depending upon whether the indicator meets a criterion, such as not being a reclaimed block.[21] Code devices are also provided for writing the data, however encoded, into the block.[22]

Independent claim 26 is directed to a memory system that includes a non-volatile memory with a plurality of blocks in an array (124; 174).[23] This memory system includes a memory area that stores various program code devices.[24] The code devices of the system of claim 26 include code devices that obtain an indicator of a number of times that a given block has been erased[25]. Code devices are provided to encode the data according to a first error detection algorithm, such as a 1-bit ECC algorithm, or a second error detection algorithm (*e.g.* a 2-bit ECC algorithm) that has a higher error detection capability than the first algorithm (*e.g.,*

---

[16] Specification, *supra,* page 26, lines 7 through 10; Figure 2b.
[17] Specification, *supra,* page 16, lines 11 through 13; Figure 2b.
[18] Specification, *supra,* page 9, lines 10 through 23; Figures 1a through 1c.
[19] Specification, *supra,* page 24, lines 1 through 3.
[20] Specification, *supra,* page 20, lines 10 through 21; Figure 5a.
[21] Specification, *supra,* page 20, line 18 through page 21, line 2; Figure 5a.
[22] Specification, *supra,* page 20, line 18 through page 21, line 2; Figure 5a..
[23] Specification, *supra,* page 9, lines 10 through 23; Figures 1a through 1c.
[24] Specification, *supra,* page 24, lines 1 through 3.
[25] Specification, *supra,* page 14, lines 9 through 26; Figure 2a.

capable of detecting two failed bits rather than one failed bit), depending upon whether the indicator meets a criterion, for example having a value below a threshold erase count.[26] Code devices are also provided for writing the data, however encoded, into the block.[27]

Independent claim 33 is also directed to a memory system that includes a non-volatile memory with a plurality of blocks in an array (124; 174),[28] and that includes a memory area that stores various program code devices.[29] The code devices of the system of claim 33 include code devices that obtain an indicator of a number of times that a given block has been erased[30], and code devices that determine whether this indicator is less than a threshold value.[31] Code devices decode the data from that block according to a first error detection algorithm, such as a 1-bit ECC algorithm, or a second error detection algorithm (*e.g.* a 2-bit ECC algorithm) that has a higher error detection capability than the first algorithm (*e.g.,* capable of detecting two failed bits rather than one failed bit), depending upon whether the indicator is less than or not less than the threshold erase count.[32]

The inventive methods and systems of this application provide important advantages in the use of semiconductor non-volatile memories, such as flash memories. As a result of this invention, a non-volatile memory can include some blocks storing data encoded according to one error detection algorithm, and other blocks storing data that is encoded according to a different error detection algorithm. Those blocks that are prone to failure (such as reclaimed blocks, or those blocks that have been erased many times) can continue to be used in the memory, but with a more stringent ECC code applied. On the other hand, those blocks that are not as prone to failure and with good expected reliability (such as blocks with low erase counts) can store data according to an ECC of lesser strength, which reduces the number of bit locations in those blocks

---

[26] Specification, *supra,* page 15, lines 6 through 16; Figure 2a.
[27] Specification, *supra,* page 15, lines 10 through 18; Figure 2a..
[28] Specification, *supra,* page 9, lines 10 through 23; Figures 1a through 1c.
[29] Specification, *supra,* page 24, lines 1 through 3.
[30] Specification, *supra,* page 15, line 20 through page 16, line 2; Figure 2b.
[31] Specification, *supra,* page 16, lines 5 through 7; Figure 2b.
[32] Specification, *supra,* page 16, lines 5 through 14; Figure 2b.

that store redundant ECC data. The memory capacity, error rate, and computational efficiency, of the flash memory system is thus optimized according to this invention.[33]

Grounds of Rejection to Be Reviewed On Appeal

The §103 rejection of claim 4 and its remaining dependent claim 3

Claims 1, 4, and 5 were finally rejected under §103 as unpatentable over the Bassett et al. reference[34] in view of the Yada et al. reference[35]. Claims 4 and 5 were also finally rejected under §103 as unpatentable over the Bassett et al. and Yada et al. references, further in view of the Bruce et al. reference[36].

In the Amendment Under Rule 116 filed December 19, 2006, claims 1 and 5 were canceled, and claim 4 was made independent, incorporating thereinto the limitations of previously presented claim 1, upon which it depended; claim 3, formerly dependent on claim 1, was amended to depend on claim 4. No change in scope was intended or presented in connection with this amendment to claim 4, and as such the bases of the final rejection stand relative to this amended claim 4.

Claim 3 was finally rejected under §103 as unpatentable over the Bassett et al. reference and Yada et al. references, further in view of Applicants' admitted prior art.

The Examiner asserted that the Bassett et al. reference teaches the encoding of data to be written to blocks of a non-volatile memory, namely a disk drive, according to more than one error correction code (ECC). Factors asserted by the Examiner as being taught by the Bassett et al. reference in this connection include the radial location of the disk to which data is to be written, media noise in a signal retrieved from the disk.[37] The Examiner admitted that the Bassett et al. reference does not expressly teach a plurality of blocks formed in a semiconductor

---

[33] Specification, *supra,* page 4, line 18 through page 5, line 3.
[34] U.S. Patent No. 6,747,827 B1, issued June 8, 2004 to Bassett et al.
[35] U.S. Patent Application Publication US 2002/0032891 A1, published March 14, 2002, from an application filed August 27, 2001.
[36] U.S. Patent No. 5,956,743, issued September 21, 1999 to Bruce et al.
[37] Office Action of September 19, 2006, pages 2 through 5.

substrate, but asserted that its disk drive teachings were applicable to such a semiconductor non-volatile memory.[38] The Examiner further asserted that the Yada et al. reference teaches the use of different ECC codes in connection with writing data to a flash memory, to maximize the efficiency with which data are stored,[39] and would therefore be properly combinable with the teachings of the Bassett et al. reference.[40] Regarding the limitation of claim 4 that the indicator is indicative of whether the block is a reclaimed block, the Examiner further asserted that the Yada et al. reference teaches this limitation in connection with its "frequently-written" parameter data.[41]

Alternatively, the Examiner asserted that the Bruce et al. reference teaches the use of erase and write counters, such erasing and rewriting corresponding to whether a block has been "reclaimed", according to claim 4.

These bases of the final rejection were maintained in the Advisory Action, despite Applicants' arguments.[42]

The additional limitation presented by claim 3, reciting that the first error detection algorithm is a 1-bit error correction code (ECC) algorithm and the second error detection algorithm is a 2-bit ECC algorithm, was found to be present in Applicants' admitted prior art.

The §103 rejection of claim 6 and its dependent claims 7 through 11 and 13

Claim 6 and its dependent claims 7, 9, and 11 were finally rejected under §103 as unpatentable over the Bassett et al. reference in view of the Yada et al. reference. Claims 7 and 8 were also finally rejected under §103 as unpatentable over the Bassett et al. and Yada et al. references, further in view of the Bruce et al. reference. Claims 9 and 10 were finally rejected under §103 as unpatentable over the Bassett et al. and Yada et al. references, further in view of

[38] Office Action, *supra,* page 5.
[39] Office Action, *supra,* page 8.
[40] Office Action, *supra,* page 9.
[41] Office Action, *supra,* page 9.
[42] Advisory Action of January 12, 2007.

the Kramer reference[43]. Claim 13 was finally rejected under §103 as unpatentable over the Bassett et al. reference and Yada et al. references, further in view of Applicants' admitted prior art.

The basis of the rejection of claim 6 was similar to that presented above relative to claim 4, with the additional assertion by the Examiner that the Yada et al. reference teaches an indicator having a value indicative of a number of times a block has been erased in connection with its "frequently-written" parameter data.[44]

Claims 7 through 9 and 11 were amended after this final rejection, to depend on claim 6 rather than upon claim 1. This amendment was entered.[45]

The §103 rejection of claim 16 and its dependent claims 17 through 20

Claims 16, 17, and 19 were finally rejected under §103 as unpatentable over the Bassett et al. reference in view of the Yada et al. reference. Claim 18 was finally rejected under §103 as unpatentable over the Bassett et al. and Yada et al. references, further in view of the Bruce et al. reference. Claims 19 and 20 were finally rejected under §103 as unpatentable over the Bassett et al. and Yada et al. references, further in view of the Kramer reference.

The specific basis of rejection of independent claim 16 was identical to that asserted against independent claim 6.[46]

The §103 rejection of claim 21 and its dependent claims 23 and 27 through 32

Claims 21 and 24 were finally rejected under §103 as unpatentable over the Bassett et al. reference in view of the Yada et al. reference.

In the Amendment Under Rule 116 filed December 19, 2006, claim 21 was amended to incorporate the limitations previously presented in claim 24, which was canceled as a result. No

[43] U.S. Patent No. 6,182,239 B1, issued January 20, 2001 to Kramer.
[44] Office Action, *supra,* page 9.
[45] Advisory Action, *supra*.
[46] *See* Office Action, *supra,* page 10 ("As to claim 16, refer to 'As to claim 6'.").

change in scope was intended or presented in connection with this amendment to claim 21, relative to claim 24. The amendment was entered.

Claims 23 and 30 were finally rejected under §103 as unpatentable over the Bassett et al. reference and Yada et al. references, further in view of Applicants' admitted prior art. Claim 27 was finally rejected under §103 as unpatentable over the Bassett et al. and Yada et al. references, further in view of the Kramer reference. Claims 28, 31, and 32 were finally rejected under §103 as unpatentable over the Bassett et al. reference and Yada et al. references.

The basis for the rejection of claim 21 was identical to that stated for claim 1, and the basis for the rejection of claim 24 was identical to that stated for claim 4.[47]

The §103 rejection of claim 26

Claim 26 was finally rejected under §103 as unpatentable over the Bassett et al. reference in view of the Yada et al. reference. The specific basis of rejection of claim 26 was identical to that asserted against independent claim 6.[48]

The §103 rejection of claim 33 and its dependent claim 34

Claim 33 was finally rejected under §103 as unpatentable over the Bassett et al. reference in view of the Yada et al. reference. Claim 34 was finally rejected under §103 as unpatentable over the Bassett et al. and Yada et al. references, further in view of the Kramer reference.

The specific basis of rejection of independent claim 33 was identical to that asserted against independent claim 6.[49]

---

[47] *See* Office Action, *supra,* page 10 ("As to claim 21, refer to 'As to claim 1'."); page 11 ("As to claim 24, refer to 'As to claim 4'.").
[48] *See* Office Action, *supra,* page 11 ("As to claim 26, refer to 'As to claim 6'.").
[49] *See* Office Action, *supra,* page 11 ("As to claim 33, refer to 'As to claim 6'.").

Argument

Claim 4 and its dependent claim 3

Appellants submit that the final rejection of claims 4 and 3 is in error, because the combined teachings of the references fall short of the requirements of the claim, and because there is no suggestion from the prior art to modify these teachings so as to reach the claims.

Appellants submit that none of the asserted references teaches or suggests the encoding of data according to a first or a second error detection algorithm, of different error detection capability, according to any attribute of the destination block in an array formed on a semiconductor substrate, much less responsive to an indicator indicative of whether that block is a reclaimed block. Appellants further submit that none of the asserted reference disclose or suggest the existence of an indicator, associated with a block of a non-volatile memory, that is indicative of whether that block has been reclaimed, much less select an error detection algorithm responsive to this indicator, as claimed.

As indicated by the Examiner, the Bassett et al. reference is directed to the error correction coding of data in a disk drive, in which different error correction strategies are applied according to the radial position of the sector to which data is to be written (for example due to media noise),[50] or according to the type of data to be written to the disk.[51] Neither of these reasons for selecting an ECC code have anything to do with any attribute of a block in a non-volatile memory having an array formed on a semiconductor substrate, as claimed. The Bassett et al. reference therefore teaches only selecting an ECC strategy according to a disk drive parameter (radial distance) that is wholly inapplicable to solid-state flash memory, or according to an attribute of the data to be written.

The Yada et al. reference, applied by the Examiner in the final rejection, also teaches only the selective ECC coding based on the nature of the data being written, and not on any past history or other attribute of the destination block. According to the Yada et al. reference:

---

[50] Bassett et al., *supra,* column 5, lines 9 through 14, and lines 26 through 33.
[51] Bassett et al., *supra,* column 6, lines 39 through 47.

> Frequently-rewritten parameter data is stored in the specified partial storage area, and program data or the like low in rewriting frequency are stored in other storage areas.[52]

with the "frequently-written parameter data" that are stored in the specified partial storage area being encoded using an ECC code, and the "program data or the like" stored in the other storage areas not receiving ECC coding.[53] This determination of whether data are to be ECC encoded is therefore based solely on the nature of the data itself, and not on any attribute of any block of the memory, much less whether a given block has been reclaimed. Therefore, the Yada et al. reference nowhere discloses encoding data according to a first or a second error detection algorithm responsive to whether an indicator indicative of whether the destination block is a reclaimed block, or responsive to any indicator indicative of any attribute of the destination block in the memory. Appellants therefore submit that the Yada et al. reference fails to meet the identifying and encoding steps of claim 4.

The Examiner also applies the Bruce et al. reference against claim 4, particularly with its teachings regarding erase/write counters in a wear-leveling scheme for a flash memory. However, the Bruce et al. memory fails to disclose using these counters to determine which of a plurality of ECC algorithms is to be applied to data to be encoded. The Bruce et al. reference therefore also necessarily fails to disclose making this determination according to an indication of whether a block is a reclaimed block, or indeed according to any attribute of the memory block.

Furthermore, Appellants submit that none of these references teaches any indicator indicative of whether a block is a reclaimed block, much less selecting an error detection algorithm in response to its value, as claimed. The Bassett et al. reference and Bruce et al. references nowhere disclose the reclaiming of blocks (either of a disk drive or of a flash memory, respectively). Nor does the Yada et al. reference disclose the existence of a "reclaimed" block. The Examiner asserted that the rewriting of a block corresponds to reclaiming a block:

---

[52] Yada et al., *supra,* paragraph [0029].
[53] Yada et al., *supra,* paragraph [0029].

> As to claim 4, Yada et al. teach that **the indicator has a value indicative of whether the block is a reclaimed block** [Frequently-rewritten parameter data is stored in the specified partial storage area, and program data or the like low in rewriting frequency are stored in other storage areas (paragraph 0029). [54]

As discussed above, the "frequently-rewritten" modifier in this passage of the reference does not pertain to the destination block whatsoever, but instead refers to the nature of the data being written (*i.e.,* "parameter data", which is frequently re-written, or "program data", which is not).[55] Secondly, the rewriting of data in a block does not "reclaim" the block, and as such a rewritten block is not a reclaimed block. The two terms have different meanings, especially as used in the specification of this application, in which substantial discussion is directed to blocks that have been erased many times (and thus have high erase counts), such discussion clearly and obviously distinct from the discussion in the specification of this application concerning the reclaiming of blocks that were previously indicated as unusable.[56] The Examiner's conclusion that an indicator of the frequency with which certain data to be written will be rewritten in the future, based on the nature of that data in connection with its user application, equates to an indicator that a physical block of a non-volatile solid-state memory has been reclaimed from its previous designation as unusable, is far beyond anything that the skilled reader of this application could imagine. Accordingly, Appellants submit that the prior art applied against claim 4 and its dependent claim 3 also lacks any teaching of an indicator that a block of the memory has been reclaimed.

Therefore, Appellants submit that the combined teachings of the applied references fall short of the requirement of claim 4 and its dependent claims, because none of the references disclose encoding data using a first or a second error detection algorithm, responsive to an indicator associated with a block of a memory formed on a semiconductor substrate either meeting or not meeting a criterion, and because none of the references disclose such an indicator indicative of whether such a block is a reclaimed block.

---

[54] Office Action, *supra,* page 9 (emphasis in original).
[55] *See also* Yada et al., *supra,* paragraphs [0114] and [0115] for the types of data considered to be frequently-written as compared with those data that are not.
[56] Specification, *supra,* page 18, line 10 through page 19, line 11.

Appellants further submit that there is no suggestion from the prior art to modify these combined teachings in such a manner as to reach claim 4. First, Appellants submit that there is no suggestion from the prior art to combine the teachings of the Bassett et al. reference, relative to its disk drive-specific factors (radial location of the sector), with the teachings of the Yada et al. and Bruce et al. references. One skilled in the art of flash memory design would simply not be motivated to use the Bassett et al. techniques in the flash memory of the Yada et al. reference, because the teachings of the reference regarding using different ECC codes for different radial locations have no connection or applicability to flash memories. Furthermore, to the extent that the Bassett et al. reference teaches using different data types,[57] the Yada et al. reference itself also refers to the selection of ECC vs. no ECC based on data type; both of these teachings fall short of the requirements of the claim. There is simply nothing in the prior art, especially that prior art cited against the claims in this case, that would suggest to the skilled reader to modify these teachings to select an error detection algorithm based on whether a physical block, to which the data is to be written, has or has not been reclaimed.

For these reasons, Appellants submit that the final rejection of claims 4 and 3 under §103 is in error, and that these claims are in fact patentable over the applied references.

Claim 6 and its dependent claims 7 through 11 and 13

Appellants submit that the final rejection of claim 6 and its dependent claims is in error, because the combined teachings of the references fall short of the requirements of the claim, and because there is no suggestion from the prior art to modify these teachings so as to reach the claims.

More specifically, Appellants submit that none of the asserted references teaches or suggests the encoding of data according to a first or a second error detection algorithm, of different error detection capability, according to any attribute of the destination block in an array formed on a semiconductor substrate, much less responsive to an indicator having a value indicative of a number of times the block has been erased, as required by claim 6.

[57] Bassett et al., *supra,* column 6, lines 39 through 47.

As discussed above and as asserted by the Examiner, the Bassett et al. reference is directed to the error correction coding of data in a disk drive, in which different error correction strategies are applied according to the radial position of the sector to which data is to be written (for example due to media noise),[58] or according to the type of data to be written to the disk.[59] Neither of these reasons for selecting an ECC code have anything to do with any attribute of a block in a non-volatile memory having an array formed on a semiconductor substrate, much less the number of times the block has been erased, as required by claim 6. Rather, the selecting of an ECC strategy according to the Bassett et al. reference is based only on a disk drive parameter (radial distance) that has nothing to do with solid-state flash memory, or according to an attribute of the data to be written itself.

The Yada et al. reference, applied by the Examiner in the final rejection, also teaches only the selective ECC coding based on the nature of the being written, and not on any past history or other attribute of the destination block. As discussed above relative to claim 4, the Yada et al. reference teaches that "frequently-written parameter data" are encoded using an ECC code and stored in a specified partial storage area, and that "program data or the like" are not ECC coded and are stored in other storage areas.[60] Specifically, the Examiner asserted that the passage:

> Frequently-rewritten parameter data is stored in the specified partial storage area, and program data or the like low in rewriting frequency are stored in other storage areas.[61]

corresponds to an indicator having a value indicative of a number of times a block has been erased. This assertion is simply wrong. The "frequently-written" modifier in this passage refers to how often one might expect that the data *will be* rewritten, based on whether the data is representative of "parameter data" (frequently rewritten) or "program data or the like" (less

[58] Bassett et al., *supra,* column 5, lines 9 through 14, and lines 26 through 33.
[59] Bassett et al., *supra,* column 6, lines 39 through 47.
[60] Yada et al., *supra,* paragraph [0029].
[61] Yada et al., *supra,* paragraph [0029].

frequently rewritten).[62] This "frequently-written" modifier in no way refers to the physical block to which this data is to be written, much less to the previous history of such a block. Instead, this asserted teaching of the Yada et al. reference pertains to the nature of the data itself.

Furthermore, in the Advisory Action, the Examiner asserted that it is well known in the art that in order for a flash memory block to be rewritten, it has to be erased first, and that there is a "one-to-one correspondence between the number of erasing and the number of rewriting of a flash block".[63] While Appellants do not necessarily agree with this statement,[64] even if one accepts this assertion for the sake of argument, this assertion is not relevant to the claim. As stated above, the Yada et al. reference looks to the frequency at which it *expects* the data to be rewritten[65]; in contrast, the claim determines the ECC code by looking to an indicator of a number of times that the block *has been* erased.

Therefore, according to the Yada et al. reference, the determination of whether data is ECC encoded is based solely on the nature of the data itself, and not on any attribute of any block of the memory, much less the number of times the block to which this data is to be written has previously been erased. Because the Yada et al. reference nowhere discloses encoding data according to a first or a second error detection algorithm responsive to whether an indicator having a value indicative of a number of times the block has been erased is less than or not less than a threshold value, Appellants therefore submit that the Yada et al. reference fails to meet the identifying and encoding steps of claim 6.

Nor do the other references cited against the dependent claims, including the Bruce et al. reference, provide such teachings.

---

[62] *See also* Yada et al., *supra,* paragraphs [0114] and [0115] for the types of data considered to be frequently-written as compared with those data that are not.
[63] Advisory Action, *supra.*
[64] For example, a block may be repeatedly written to page by page with updated data. A logical page of data may be written at one time to page 1 of a given block, and an updated version of that logical page of data may later be written to page 2 of that same block. In this sense, data in that block can be "rewritten" without the block being erased between each write operation.
[65] *See also* Yada et al., *supra,* paragraphs [0114] and [0115] for the types of data considered to be frequently-written as compared with those data that are not.

Therefore, Appellants submit that the combined teachings of the applied references fall short of the requirement of claim 6 and its dependent claims.

Appellants further submit that there is no suggestion from the prior art to modify these combined teachings in such a manner as to reach claim 6 and its dependent claims. Appellants first submit, as before, that there is no suggestion from the prior art to combine the teachings of the Bassett et al. reference, relative to its disk drive-specific factors (radial location of the sector) with the teachings of the Yada et al. and Bruce et al. references. One skilled in the art of flash memory design would simply not be motivated to use the Bassett et al. techniques in the flash memory of the Yada et al. reference, because the teachings of the reference regarding using different ECC codes for different radial locations have no connection or applicability to flash memories.[66] Nor is there any suggestion from the Bassett et al. and Yada et al. references beyond selecting between ECC and no ECC based on data type, as taught by Yada et al. Rather, the references give no hint that one ought to look at the erase history of the block to determine the error correction algorithm. Nothing in the prior art, especially that prior art cited against the claims in this case, suggests to the skilled reader to modify its teachings to select an error detection algorithm based on an indicator of an erase count for the destination block.

For these reasons, Appellants submit that the final rejection under §103, of claims 6 through 11 and 13, is in error and should be reversed.

Claim 16 and its dependent claims 17 through 20

Appellants submit that the final rejection of claim 16 and its dependent claims is in error, because the combined teachings of the references fall short of the requirements of the claim, and because there is no suggestion from the prior art to modify these teachings so as to reach the claims.

More specifically, Appellants submit that none of the asserted references teaches or suggests the decoding of data according to a first or a second error detection algorithm, of

[66] Bassett et al., *supra,* column 6, lines 39 through 47.

different error detection capability, according to any attribute of the block in an array formed on a semiconductor substrate, from which the data is being read, much less responsive to an indicator having a value indicative of a number of times that block has been erased, as required by claim 16.

As pointed out repeatedly above, the selecting of an ECC strategy according to the Bassett et al. reference is based only on a disk drive parameter (radial distance) that is wholly inapplicable to solid-state flash memory, or based on an attribute of the data itself. Nothing in the Bassett et al. reference teaches the selecting of an ECC strategy according to a number of times that the block to be read has been erased, even in the context of its disk drive.

The Yada et al. reference teaches only the selective ECC coding[67] based on the nature of the data itself, rather than on the past history or other attribute of the block from which data is to be read. As discussed above, the Examiner cites the passage:

> Frequently-rewritten parameter data is stored in the specified partial storage area, and program data or the like low in rewriting frequency are stored in other storage areas.[68]

as teaching an indicator that has a value indicative of a number of times a block has been erased. Appellants submit that this assertion is simply wrong. The "frequently-written" modifier in this passage refers to how often one might expect that the data *will be* rewritten, based on whether the data is representative of "parameter data" (frequently rewritten) or "program data or the like" (less frequently rewritten). While, in the Advisory Action, this assertion is amplified into the statement that there is a "one-to-one correspondence between the number of erasing and the number of rewriting of a flash block",[69] Appellants submit that this assertion is misplaced, because the reference is concerned with the nature of the data and expectations regarding that data, and not with any attribute of the memory from which that data will be read. Appellants therefore submit that the Yada et al. reference nowhere discloses decoding data according to a first or a second error detection algorithm responsive to whether an indicator having a value

[67] And, presumably, decoding.
[68] Yada et al., *supra,* paragraph [0029].

indicative of a number of times that the block, from which the data is to be read, has been erased is less than or not less than a threshold value. Appellants therefore submit that the Yada et al. reference fails to meet the determining and decoding steps of claim 16.

Nor do the other references cited against the dependent claims, including the Bruce et al. and Kramer references, provide such teachings.

Therefore, Appellants submit that the combined teachings of the applied references fall short of the requirement of claim 16 and its dependent claims.

Appellants further submit that there is no suggestion from the prior art to modify these combined teachings in such a manner as to reach claim 16 and its dependent claims. Appellants again submit that there is no suggestion from the prior art to combine the disk drive-specific factors taught by the Bassett et al. reference with the teachings of the Yada et al. and Bruce et al. references. Nor is there any suggestion from the Bassett et al. and Yada et al. references to select between ECC and no ECC, or among ECC codes or strategies, other than based on the type or nature of the data, as taught by Yada et al. There is no hint from this prior art that one ought to look at the erase history of the block to determine the error correction algorithm to be applied in decoding data read from that block.

For these reasons, Appellants submit that the final rejection, under §103, of claims 16 through 20 is in error and should be reversed.

Claim 21 and its dependent claims 23, and 27 through 32

Appellants submit that the final rejection of claims 21 and its dependent claims is in error, because the combined teachings of the references fall short of the requirements of the claim, and because there is no suggestion from the prior art to modify these teachings so as to reach the claims.

---

[69] Advisory Action, *supra.*

More specifically, Appellants submit that none of the asserted references teaches or suggests code devices for encoding of data according to a first or a second error detection algorithm, of different error detection capability, according to any attribute of the destination block in an array formed on a semiconductor substrate, much less responsive to an indicator indicative of whether that block is a reclaimed block. Appellants further submit that none of the asserted reference disclose or suggest the existence of an indicator, associated with a block of a non-volatile memory, that is indicative of whether that block has been reclaimed, much less responsive to which code devices encode data according to an error detection algorithm selected responsive to such an indicator.

As previously argued, Appellants submit that the Bassett et al. reference is directed to the error correction coding of data in a disk drive, in which different error correction strategies are applied according to the radial position of the sector to which data is to be written (for example due to media noise),[70] or according to the type of data to be written to the disk.[71] There is no mention in this reference of any circuitry or code devices that encode data with a first or a second error detection algorithm, selected responsive to any attribute of a non-volatile memory block in an array formed on a semiconductor substrate, to which the data are to be written.

Also as previously argued, the Yada et al. reference also lacks teachings in this regard, because it teaches only the selection of whether to apply ECC coding in response to the nature of the data being written, and not on any past history or other attribute of the destination block. The Examiner asserts that Yada et al. meet this claim limitation by way of its "frequently-written parameter data" that are stored in the specified partial storage area being encoded using an ECC code, and the "program data or the like" stored in the other storage areas not receiving ECC coding.[72] Appellants disagree. The Yada et al. determination of whether data are to be ECC encoded is based solely on the nature of the data itself, and not on any attribute of any block of the memory, much less whether a given block has been reclaimed. Appellants therefore submit that the Yada et al. reference nowhere discloses or suggests code devices for encoding data

[70] Bassett et al., *supra,* column 5, lines 9 through 14, and lines 26 through 33.
[71] Bassett et al., *supra,* column 6, lines 39 through 47.
[72] Yada et al., *supra,* paragraph [0029].

according to a first or a second error detection algorithm responsive to whether an indicator indicative of whether the destination block is a reclaimed block, and therefore also falls short of the requirements of claim 21.

The Bruce et al. reference adds no teachings to the Bassett et al. and Yada et al. references in this regard.

In addition, Appellants submit that none of these references teaches any indicator indicative of whether a block is a reclaimed block, much less code devices for encoding data using an error detection algorithm determined in response to its value, as claimed. The Bassett et al. reference and Bruce et al. references nowhere disclose the reclaiming of blocks (either of a disk drive or of a flash memory, respectively). Nor does the Yada et al. reference disclose the existence of a "reclaimed" block, even in connection with its description of "frequently-rewritten" data.[73] As discussed above, the Yada et al. reference uses the "frequently-rewritten" modifier in connection with the data to be written to the memory, but not in connection with any attributes or past history of the destination block for this data. Specifically, the reference describes "parameter data" as frequently re-written, and "program data and the like" as not frequently rewritten.[74] Secondly, as also discussed above, the rewriting of data in a block does not "reclaim" the block, and as such a rewritten block is not a reclaimed block. These two terms have different meanings, especially as used in the specification of this application, in which substantial discussion is directed to blocks that have been erased many times (and thus have high erase counts), such discussion clearly and obviously distinct from the discussion in the specification of this application concerning the reclaiming of blocks that were previously indicated as unusable.[75] The Examiner is incorrect in asserting that an indicator of the frequency with which certain data to be written will be rewritten in the future, based on the nature of that data in connection with its user application, equates to an indicator that a physical block of a non-volatile solid-state memory has been reclaimed from its previous designation as unusable.

---

[73] Yada et al., *supra,* paragraph [0029].
[74] *See also* Yada et al., *supra,* paragraphs [0114] and [0115] for the types of data considered to be frequently-written as compared with those data that are not.
[75] Specification, *supra,* page 18, line 10 through page 19, line 11.

Appellants therefore submit that the prior art applied against claim 21 and its dependent claims also falls short of the claims, because this prior art fails to teach or suggest an indicator that a block of the memory has been reclaimed, much less use of that indicator by code devices for encoding data that is later written to the block.

For these reasons, Appellants submit that the combined teachings of the applied references fall short of the requirement of claim 21 and its dependent claims.

Appellants further submit that there is no suggestion from the prior art to modify these combined teachings in such a manner as to reach claim 21 and its dependent claims. First, Appellants submit that there is no suggestion from the prior art to combine the teachings of the Bassett et al. reference, relative to its disk drive-specific factors (radial location of the sector) with the teachings of the Yada et al. and Bruce et al. references. One skilled in the art of flash memory design would simply not be motivated to use the Bassett et al. techniques in the flash memory of the Yada et al. reference, because the teachings of the reference regarding using different ECC codes for different radial locations have no connection or applicability to flash memories. Furthermore, to the extent that the Bassett et al. reference teaches using different data types,[76] the Yada et al. reference itself also refers to the selection of ECC vs. no ECC based only on data type. Neither of these references, nor any other reference in the prior art, suggests to the skilled reader to modify the properly combined teachings of the Bassett et al. and Yada et al. references to provide a memory system with code devices for encoding data according to a first or second error detection algorithm, responsive to an indicator indicative of whether a physical block has been reclaimed.

For these reasons, Appellants submit that the final rejection under §103 of claim 21 and its dependent claims is in error, and that these claims are in fact patentable over the applied references.

[76] Bassett et al., *supra,* column 6, lines 39 through 47.

Claim 26

Appellants submit that the final rejection of claim 26 is in error, because the combined teachings of the references fall short of the requirements of the claim, and because there is no suggestion from the prior art to modify these teachings so as to reach the claim.

More specifically, Appellants submit that none of the asserted references teaches or suggests code devices for encoding data according to a first or a second error detection algorithm, of different error detection capability, according to any attribute of the destination block of that data, that block being in an array formed on a semiconductor substrate, much less such code devices that encode the data responsive to an indicator having a value indicative of a number of times that block has been erased, as required by claim 26.

As pointed out repeatedly above, the selecting of an ECC strategy according to the Bassett et al. reference is based only on a disk drive parameter (radial distance) that is wholly inapplicable to solid-state flash memory, or based on an attribute of the data itself. Nothing in the Bassett et al. reference teaches the selecting of an ECC encoding strategy according to a number of times that the destination block has been erased, even in the context of its disk drive.

The Yada et al. reference teaches only the selective ECC coding based on the nature of the data itself, rather than on the past history or other attribute of the block from which data is to be read. As discussed above, the Examiner cites the passage:

> Frequently-rewritten parameter data is stored in the specified partial storage area, and program data or the like low in rewriting frequency are stored in other storage areas.[77]

as teaching an indicator that has a value indicative of a number of times a block has been erased. Appellants submit that this assertion is simply wrong. The "frequently-written" modifier in this passage refers to how often one might expect that the data *will be* rewritten, based on whether the data is representative of "parameter data" (frequently rewritten) or "program data or the like" (less frequently rewritten). As such, the asserted teaching of the Yada et al. reference pertains to

the nature of the data itself. Appellants therefore submit that the Yada et al. reference nowhere discloses code devices for encoding data according to a first error detection algorithm responsive an indicator having a value indicative of a number of times that the block, to which the data is to be written, has been erased being less than a threshold value, or code devices for encoding data according to a second error detection algorithm responsive to that indicator being not less than the threshold value. Appellants therefore submit that the Yada et al. reference fails to meet the code devices for encoding required by claim 26.

The Examiner stated, in the Advisory Action, that there is a "one-to-one correspondence between the number of erasing and the number of rewriting of a flash block",[78] Even accepting this statement for the sake of argument,[79] Appellants submit that this assertion remains misplaced, because even this interpretation of the reference is concerned only with the nature of the data and expectations regarding future actions upon that data to be written, and not with any attribute concerning the past history of the memory block to which that data is to be written.

Nor do the other references cited against the dependent claims, including the Bruce et al. and Kramer references, provide such teachings.

Therefore, Appellants submit that the combined teachings of the applied references fall short of the requirement of claim 26.

Appellants further submit that there is no suggestion from the prior art to modify these combined teachings in such a manner as to reach claim 26. Appellants again submit that there is no suggestion from the prior art to combine the disk drive-specific factors taught by the Bassett et al. reference with the teachings of the Yada et al. and Bruce et al. references. Nor is there any suggestion from the Bassett et al. and Yada et al. references to select between ECC and no ECC, or among ECC codes or strategies, other than based on the type or nature of the data, as taught by Yada et al. There is no hint from this prior art that one ought to look at the erase history of the block to determine the error correction algorithm to be applied to encode data prior to writing

---

[77] Yada et al., *supra,* paragraph [0029].
[78] Advisory Action, *supra.*

the data to the destination block, nor provide a system with code devices for accomplishing those functions, as required by claim 26.

For these reasons, Appellants submit that the final rejection of claim 26, under §103, is in error and should be reversed.

Claim 33 and its dependent claim 34

Appellants submit that the final rejection of claim 33 and its dependent claim 34 is in error, because the combined teachings of the references fall short of the requirements of the claims, and because there is no suggestion from the prior art to modify these teachings so as to reach the claims.

Similarly as argued above, Appellants submit that none of the asserted references teaches or suggests code devices for decoding data according to a first or a second error detection algorithm, of different error detection capability, according to any attribute of the block in an array formed on a semiconductor substrate, from which the data is being read, much less responsive to an indicator having a value indicative of a number of times that block has been erased, as required by claim 33.

As pointed out repeatedly above, the selecting of an ECC strategy according to the Bassett et al. reference is based only on a disk drive parameter (radial distance) that is wholly inapplicable to solid-state flash memory, or based on an attribute of the data itself. Nothing in the Bassett et al. reference teaches the selecting of an ECC strategy according to a number of times that the block to be read has been erased, even in the context of its disk drive.

The Yada et al. reference teaches only the selective ECC coding[80] based on the nature of the data itself, rather than on the past history or other attribute of the block from which data is to be read. As discussed above, the Examiner cites the passage:

---

[79] Appellants do not admit that the statement is accurate, as discussed above.
[80] And, presumably, decoding.

> Frequently-rewritten parameter data is stored in the specified partial storage area, and program data or the like low in rewriting frequency are stored in other storage areas.[81]

as teaching an indicator that has a value indicative of a number of times a block has been erased. Appellants submit that this assertion is simply wrong. The "frequently-written" modifier in this passage refers to how often one might expect that the data *will be* rewritten, based on whether the data is representative of "parameter data" (frequently rewritten) or "program data or the like" (less frequently rewritten). While, in the Advisory Action, this assertion is amplified into the statement that there is a "one-to-one correspondence between the number of erasing and the number of rewriting of a flash block",[82] Appellants submit that this assertion remains misplaced, because the reference (even as interpreted by the Examiner) is concerned solely with the nature of the data and expectations regarding the rewriting of that data, and not with any attribute of the memory from which that data will be read. Appellants therefore submit that the Yada et al. reference nowhere discloses code devices for decoding data according to a first or a second error detection algorithm responsive to whether an indicator having a value indicative of a number of times that the block, from which the data is to be read, has been erased is less than or not less than a threshold value. Appellants therefore submit that the Yada et al. reference fails to meet element of the code devices for decoding, required by claim 33.

Nor do the other references cited against the dependent claims, including the Bruce et al. and Kramer references, provide such teachings.

Appellants therefore submit that the combined teachings of the applied references fall short of the requirement of claims 33 and 34.

Appellants further submit that there is no suggestion from the prior art to modify these combined teachings in such a manner as to reach claim 33 and its dependent claim 34. Appellants again submit that there is no suggestion from the prior art to combine the disk drive-specific factors taught by the Bassett et al. reference with the teachings of the Yada et al. and

---

[81] Yada et al., *supra,* paragraph [0029].
[82] Advisory Action, *supra.*

Bruce et al. references. Nor is there any suggestion from the Bassett et al. and Yada et al. references to select between ECC and no ECC, or among ECC codes or strategies, other than based on the type or nature of the data, as taught by Yada et al. There is no hint from this prior art that one ought to look at the erase history of the block to determine the error correction algorithm to be applied in decoding data read from that block, much less to provide code devices for decoding data read from a block in response to such an indicator of the number of times the block has been erased.

For these reasons, Appellants submit that the final rejection, under §103, of claims 33 and 34 is in error and should be reversed.

In conclusion

For the above reasons, Applicants respectfully submit that the final rejection of the claims on appeal is in error, and should be reversed. Reversal of that final rejection is respectfully requested.

Respectfully submitted,

/Rodney M. Anderson/

Rodney M. Anderson
Registry No. 31,939
Attorney for Appellants

Anderson, Levine & Lintel, L.L.P.
14785 Preston Road, Suite 650
Dallas, Texas 75254
(972) 664-9554

Claims appendix:

3. The method of claim 4 wherein the first error detection algorithm is a 1-bit error correction code (ECC) algorithm and the second error detection algorithm is a 2-bit ECC algorithm.

4. A method for storing data within a non-volatile memory comprised of a plurality of blocks in an array formed on a semiconductor substrate, each of the plurality of blocks having an indicator indicative of whether the block is a reclaimed block, the method comprising:

identifying a first block of the plurality of blocks into which the data is to be stored;

responsive to the indicator associated with the first block meeting a criterion, encoding the data using a first error detection algorithm;

then writing the encoded data into the first block;

identifying a second block of the plurality of blocks into which data is to be stored;

responsive to the indicator associated with the second block not meeting the criterion, encoding the data using a second error detection algorithm, the second error detection algorithm having a higher error detection capability than the first error detection algorithm; and

then writing the encoded data into the second block.

6. A method for storing data within a non-volatile memory, comprised a plurality of blocks in an array formed on a semiconductor substrate, of a memory system, the method comprising:

identifying one of the plurality of blocks into which the data is to be stored;

obtaining an indicator associated with the identified block, the indicator having a value indicative of a number of times the first block has been erased;

determining whether the indicator is less than a threshold value,

responsive to the indicator being less than the threshold value, encoding the data using the first algorithm and then writing the data encoded using a first error detection algorithm into the identified block;

responsive to the indicator not being less than the threshold value, encoding the data using a second algorithm and then writing the data encoded using a second error detection algorithm into the identified block, the second error detection algorithm having a higher error detection capability than the first error detection algorithm;

repeating the identifying, obtaining, determining, encoding, and writing steps for another block in the array;

wherein, as a result of the repeating step, a first block in the array stores data encoded according to the first algorithm, and a second block in the array stores data encoded according to the second algorithm.

7. The method of claim 6 wherein the indicator has a value indicative of an approximately average number of times blocks within the non-volatile memory have been erased.

8. The method of claim 6 wherein the indicator is stored in a data structure, the data structure being substantially separate from the first block,

and wherein obtaining the indicator associated with the block includes obtaining the indicator from the data structure.

9. The method of claim 6 wherein the non-volatile memory is a flash memory.

10. The method of claim 9 wherein the flash memory is one of a NAND flash memory and an MLC NAND flash memory.

11. The method of claim 6, further comprising:

identifying the first block as a block from which data is to be read;

obtaining the indicator associated with the first block;

responsive to the indicator associated with the first block meeting a criterion, decoding the data stored in the first block using the first error detection algorithm; and

responsive to the indicator associated with the first block not meeting the criterion, decoding the data using the second error detection algorithm.

13. The method of claim 11 wherein the first error detection algorithm is a 1-bit ECC algorithm and the second error detection algorithm is a 2-bit ECC algorithm.

16. A method for reading data within a non-volatile memory comprised of a plurality of blocks in an array formed on a semiconductor substrate, the method comprising:

identifying one of the plurality of blocks from which data is to be read;

obtaining an indicator associated with the identified block, the indicator having a value indicative of a number of times the identified block has been erased;

determining whether the indicator is less than a threshold value,

responsive to the indicator being less than the threshold value, decoding the data using a first error detection algorithm;

responsive to the indicator not being less than the threshold value, decoding the data using a second error detection algorithm, the second error detection algorithm having a higher error detection capability than the first error detection algorithm.

17. The method of claim 11 wherein the indicator has a value indicative of an approximately average number of times physical blocks of the non-volatile memory have been erased.

18. The method of claim 11 wherein the indicator is stored in a data structure, the data structure being substantially separate from the first block,

and wherein obtaining the indicator associated with the block includes obtaining the indicator from the data structure.

19. The method of claim 11 wherein the non-volatile memory is a flash memory.

20. The method of claim 19 wherein the flash memory is one of a NAND flash memory and an MLC NAND flash memory.

21. A memory system comprising:

a non-volatile memory array formed on a semiconductor substrate, the array including a plurality of blocks;

code devices for identifying a block into which data is to be stored;

code devices for obtaining an indicator associated with the identified block, the indicator having a value arranged to indicate whether the block is a reclaimed block;

code devices for encoding the data using a first error detection algorithm responsive to the indicator meeting a criterion, and for encoding the data using a second error detection algorithm responsive to the indicator not meeting the criterion, the second error detection algorithm having a higher error detection capability than the first error detection algorithm;

code devices for writing encoded data into the identified block; and

a memory area that stores the code devices.

23. The memory system of claim 21 wherein the first error detection algorithm is a 1-bit ECC algorithm and the second error detection algorithm is a 2-bit ECC algorithm.

26. A memory system comprising:

a non-volatile memory array formed on a semiconductor substrate, the array including a plurality of blocks;

code devices for identifying a block into which data is to be stored;

code devices for obtaining an indicator associated with the identified block, the indicator having a value indicative of a number of times the identified block has been erased;

code devices for determining whether the indicator is less than a threshold value,

code devices for encoding the data using a first algorithm responsive to the indicator being less than the threshold value;

code devices for writing the data encoded using the first algorithm into the identified block;

code devices for encoding the data using a second algorithm responsive to determining that the data is not to be encoded using the first algorithm, the second error detection algorithm having a higher error detection capability than the first error detection algorithm;

code devices for writing the data encoded using the second algorithm into the identified block; and

a memory area that stores the code devices.

27. The memory system of claim 21 wherein the non-volatile memory is one of a NAND flash memory and an MLC NAND flash memory.

28. The memory system of claim 21, further comprising:

code devices for determining whether the indicator meets the criterion;

code devices for decoding the data using the first error detection algorithm responsive to the indicator meeting the criterion, and for decoding the data using the second error detection algorithm responsive to the indicator not meeting the criterion.

30. The memory system of claim 28 wherein the first error detection algorithm is a 1-bit ECC algorithm and the second error detection algorithm is a 2-bit ECC algorithm.

31. The memory system of claim 28 wherein the indicator is arranged to indicate whether the block is a reclaimed block.

32. The memory system of claim 28 wherein the indicator has a value indicative of a number of times the block has been erased.

33. A memory system comprising:

a non-volatile memory array formed on a semiconductor substrate, the array including a plurality of blocks, each including data;

code devices for identifying a block;

code devices for obtaining an indicator associated with the identified block, the indicator having a value indicative of a number of times the block has been erased;

code devices for determining whether the indicator is less than a threshold value,

code devices for decoding the data using a first error detection algorithm responsive to the indicator being less than the threshold value;

code devices for decoding the data using a second error detection algorithm responsive to the indicator not being less than the threshold value, the second error detection algorithm having a higher error detection capability than the first error detection algorithm; and

a memory area that stores the code devices.

34. The memory system of claim 28 wherein the non-volatile memory is one of a NAND flash memory and an MLC NAND flash memory.

Evidence appendix:

None.

Related proceedings appendix:

None.